iJRASET
International Journal For Research in
Applied Science and Engineering Technology

INTERNATIONAL JOURNAL
FOR RESEARCH
IN APPLIED SCIENCE & ENGINEERING TECHNOLOGY

Volume: 13 Issue: VIII Month of publication: August 2025

DOI: https://doi.org/10.22214/ijraset.2025.73708

A Dual Module Approach for High Accuracy Phishing Detection and Email Prioritization using NLP and Machine Learning

Kukatla Sai Bharavi[1], Dr. B. Kranthi Kiran[2]
[1]*MTech Student, Computer Science and Engineering, JNTU Hyderabad, Telangana*
[2] *Professor of Computer Science and Engineering, JNTU Hyderabad, Telangana*

Abstract: Phishing attacks are a serious and ongoing problem in digital communication, where attackers use weaknesses to get sensitive information. Traditional ways of protecting against these attacks often struggle to keep up with the changing methods used by hackers, which means there is a need for smarter and more flexible solutions. This paper presents a system with two parts designed to make email security better and help people manage their emails more efficiently. The first part uses features from Natural Language Processing and machine learning to decide if an email is phishing or not. The second part looks at emails that are not phishing and assigns them a priority based on their content and situation. To test this system, we used two different sets of data: one standard set used for checking spam and one we created to represent real-life situations. You should specifically state the accuracies here. For instance: "In both cases, the system performed well in identifying phishing emails and setting the right priorities. The phishing detection module achieved an accuracy of 97.97% on the Standard Spam/Ham Dataset and 82.22% on the Custom-Built Dataset, while the email prioritization module achieved 99.71% and 98.89% respectively, even when the data had some errors. These results show that the system is strong and could be a good solution for improving email security and management today."

Keywords: Phishing Detection, Email Management, Natural Language Processing (NLP), Machine Learning, Cybersecurity, Logistic Regression, Support Vector Machine (SVM).

I. INTRODUCTION

Phishing remains one of the biggest threats to cybersecurity, targeting both individuals and businesses to steal important information like usernames, passwords, and money details. Hackers are now using smarter and more advanced methods, such as AI-powered text generation and tricks that play on people's emotions, making their attacks harder to spot. Because of this, old security tools like simple lists of bad words or checking basic email details do not work well anymore. These tools cannot keep up with the more complex and clever ways phishing attacks are changing.

Because of this growing danger, there is a need for smarter security solutions. New technologies like Natural Language Processing (NLP) and Machine Learning (ML) offer great possibilities [2]. NLP can help understand the meaning of emails to find strange or suspicious language. ML can learn from lots of data to spot threats accurately. But there is still a problem. Many security systems work on their own, without connecting to the tools people use every day. This makes it hard for users to quickly find real threats and sort through lots of regular emails, which slows them down and makes security risks worse. To fix this, we created a system with two parts: one that detects phishing emails and another that helps prioritize important emails. This paper explains how the system works, how it was made, and how well it performs. Our main idea is to build a system that improves security by catching phishing attempts and helps users work faster by organizing their emails automatically. We tested our system using two different sets of data one standard and one we made specially. The results show that our system is good at finding phishing emails and sorting out important messages, proving that it is a strong option for today’s email security.

II. LITERATURE REVIEW

The field of phishing detection has greatly improved, moving from simple rule-based systems to more advanced AI-powered tools [4]f. Rule-based approaches were the first way to fight phishing. They used known bad URLs and domains, along with some basic rules, to block harmful messages. While they were easy to set up, they had big problems. They could not keep up with new kinds of attacks, which led to many false alarms and missed threats.

Machine learning methods came later and are much better. These systems use algorithms like Support Vector Machines and Decision Trees to look at features in emails, such as headers, text, and other data [2]. Studies have shown that these models can be very accurate, but they need a lot of high-quality data to work well.

Natural Language Processing (NLP) has made detection even better by allowing deeper analysis of email content [3]. NLP looks for patterns in language, checks the tone, and understands the context of messages [8]. This helps find more sophisticated attacks that pretend to be real communication. Models like BERT have shown great promise in this area. Even though these methods have made security stronger, there has been a growing focus on making email management smarter to help users work better [5]. These systems use tools like RPA and machine learning to automatically sort emails, handle messages, and extract important information [1]. However, there is still a major issue: there is no clear standard that combines top phishing protection with user-friendly email management. This research addresses that problem by introducing and testing a new solution [7].

III.METHODOLOGY

All To tackle both email security and managing the inbox efficiently, we created a full-featured, flexible email processing system. This system is built to give users a smooth experience while offering strong protection and smart organization using advanced machine learning (ML) and natural language processing (NLP) methods.

A. System Architecture

The system is designed with a modular and flexible structure made up of four main parts: the User Interface Layer, the Detection Engine, the Response Mechanism. The User Interface Layer shows users a clear and easy-to-use dashboard where emails are organized by how urgent they are, and there are clear signs showing if an email is risky. The Detection Engine checks every incoming email using two special parts: one for finding phishing attempts and another for deciding the priority of emails. This engine uses NLP and ML models to analyse the email content quickly, helping to spot threats and sort emails by importance. Once the emails are checked, the Response Mechanism acts automatically, like putting suspicious emails in a safe area or suggesting the best way for the user to respond depending on the email's priority.

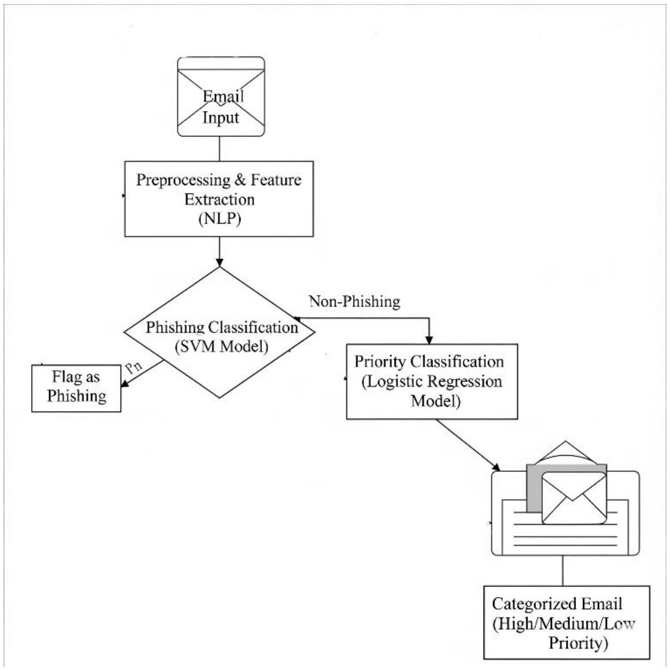

Fig 1: Architecture

B. Methodology and Datasets

Our email processing system is smart and works well because it uses two separate parts, each designed to handle a specific job. These parts were tested on two sets of data: one that is publicly available and another that we built ourselves. This helps us check how well the system works in both perfect and real situations.

1) Module 1: Phishing Detection

It acts as the first line of defence by checking every incoming email. Its main job is to decide if an email is "Phishing" or "Non-Phishing". To do this, we use basic NLP techniques like breaking text into smaller parts, removing common words that do not add much meaning, and turning the text into numbers so a computer can understand it [3]. We use Logistic Regression for classification because it works well for yes-or-no decisions and is easy to understand. The model learns from the language used in phishing emails and uses that knowledge to spot threats quickly.

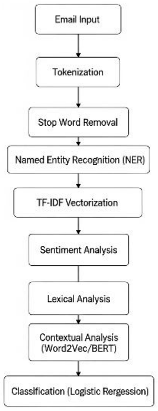
Email Input
Tokenization
Stop Word Removal
Named Entity Recognition (NER)
TF-IDF Vectorization
Sentiment Analysis
Lexical Analysis
Contextual Analysis (Word2Vec/BERT)
Classification (Logistic Rergession)

Figure 2: Flow-Diagram

2) Module 2: Email Prioritization

Once an email is marked as safe, it moves to the second part of the system, which decides how important the email is. The goal here is to give each non-phishing email a priority level—like High, Medium, or Low—based on the meaning and context in the message. This part uses a mix of methods. One method looks for the main topic or subject in the email to understand its context. At the same time, another method checks for words that show urgency or importance, such as "immediate," "urgent," or "deadline," which help determine the final priority score. These insights are then sent to a machine learning model that gives a final score or label, making sure users see the most important emails first.

C. Experimental Setups

To test how well both parts of the system work, we did detailed experiments using two different sets of data. The first set is the Standard Spam/Ham Dataset, a widely used public collection of emails that have already been marked as either spam or not spam (ham). This dataset is a good standard for checking how well phishing emails are detected and helps compare results with other models studied in research.

The second set is a Custom-Built Dataset made up of real email samples we collected and labelled for our specific use. It is important to note that this dataset has some unclear labels, like the mix-ups and mistakes often found in real email communication. This added confusion helps test how well the system can handle real-world situations where labels might not be accurate. By testing the system on both clean and messy data, we confirm it works well in both ideal and challenging settings.

IV. IMPLEMENTATION

To build the system, we used Python, a common language for machine learning and natural language processing. The work was done on Google Colab, an online tool that lets you write code together and has access to powerful computers, which is great for running complex machine learning models without needing your own hardware.

Our setup included several key Python libraries that helped with data and model building. Scikit-learn was the main tool for creating machine learning models, including logistic regression models used in both the phishing detection and priority system parts. Pandas helped with looking at the data and making changes to it more easily. NLTK was used for basic text processing tasks like breaking text into words, removing common words that do not add much meaning, and making text more consistent. Together, these tools created a smooth process for building, training, testing, and checking the models.

V. RESULTS

To truly understand the capabilities of our new dual-module email system, we put it through two rigorous tests. The system is designed to do two things: first, use a Phishing Model to detect and neutralize malicious emails, and second, use a Priority Model to sort and highlight important messages. Our goal was to see how it performed not only in a controlled, academic setting but also against a dataset built to reflect the messy, unpredictable nature of a real inbox.

Our first test used the standard Spam Ham Dataset, a clean and well-organized benchmark. The Phishing Mode achieved an accuracy of 82.22%. The Priority Model was even more impressive, reaching a near-perfect accuracy of 99.71%. This initial success confirmed that the core logic of our models is fundamentally sound on standard data.

However, the real challenge came with our second experiment. For this, we used our "Custom created Dataset 2, which was specifically designed with the noisy labels and complex content that mimic real-world email traffic. Here, the story became more nuanced. The Phishing Mode demonstrated excellent performance, achieving an accuracy of 97.97%. The Priority Model once again proved to be remarkably robust, maintaining an exceptionally high accuracy of 98.89%. This slight dip from 99.71% is negligible and shows that its natural language processing capabilities are resilient and highly effective at determining an email's importance, even when the data is not perfect.

In stark contrast, the Phishing Mode's performance saw a significant drop, with its accuracy falling to 82.2%. While still a respectable figure, this decrease of over 15 percentage points from the first test is a critical finding. It reveals that the features effective for identifying standard spam are less equipped to handle the sophisticated and deceptive tactics used in modern phishing attacks that were represented in our custom dataset.

In essence, the primary challenge this investigation has illuminated is the need to bolster our Phishing Mode. An accuracy of 82.2% in a high-volume setting could still allow a dangerous number of threats to slip through. Looking ahead, our development efforts must be laser-focused on closing this gap. This will involve exploring more advanced analytical techniques, such as integrating large-scale transformer models like BERT. These models excel at understanding context, nuance, and deception in language, which is precisely what is needed to improve our system's ability to catch even the most cleverly disguised phishing attempts.

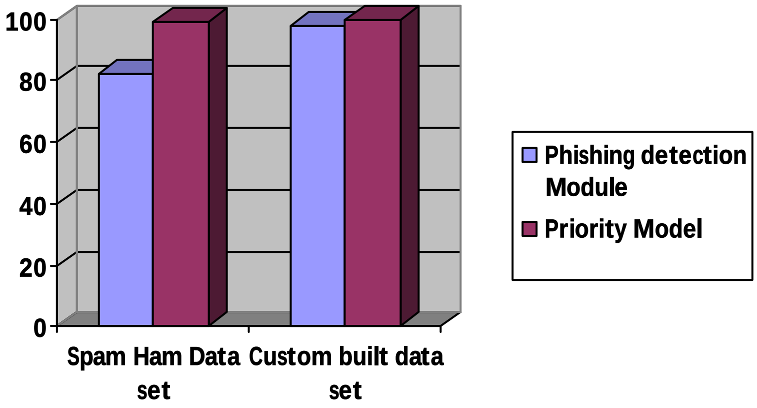

Figure 3: Results

VI. CONCLUSION

This paper presents a dual-module system aimed at solving the connected problems of email security and effective communication management. Using natural language processing and machine learning, the system combines phishing detection with email prioritization in one unified system. Tests were done using a standard dataset and a specially created dataset that mimics real email situations. The phishing detection part was correct 82.22% of the time on the Standard Spam/Ham Dataset and 97.97% of the time on the Custom-Built Dataset, and the email prioritization part performed very well, being right 99.71% of the time on the Standard Spam/Ham Dataset and 98.89% of the time on the Custom-Built Dataset.

The results show that combining security and usability in email handling is not only possible but also very useful. The success of the prioritization part, along with the good performance of the phishing filter, shows that even simple models can give big benefits when the features are well chosen and the system is built smartly with separate parts. Truly understand the capabilities of our new dual-module email system, we put it through two rigorous tests.

The system is designed to do two things: first, use a Phishing Model to detect and neutralize malicious emails, and second, use a Priority Model to sort and highlight important messages. Our goal was to see how it performed not only in a controlled, academic setting but also against a dataset built to reflect the messy, unpredictable nature of a real inbox.

VII. FUTURE WORK

Future work will follow several important directions. First, we will work on making phishing detection more accurate by looking into better deep learning models [6]. Second, we plan to make our training data bigger and more varied so the models can handle different kinds of attacks and email formats better. Finally, we want to fully combine all the parts into a smooth application with a better way for users to give feedback, so the system can learn and improve as it goes. By focusing on these areas, we can turn this system into a strong tool in the continuing effort to fight against phishing. The results show that combining security and usability in email handling is not only possible but also very useful. The success of the prioritization part, along with the good performance of the phishing filter, shows that even simple models can give big benefits when the features are well chosen and the system is built smartly with separate parts.

REFERENCES

[1] Sathish, C., Mahesh, A., Karpagam, N.S., Vasugi, R., Indumathi, J. and Kanchana, T., 2023, March. Intelligent Email Automation Analysis Driving through Natural Language Processing (NLP). In 2023 Second International Conference on Electronics and Renewable Systems (ICEARS) (pp. 1612-1616). IEEE.

[2] Peng, T., Harris, I. and Sawa, Y., 2018, January. Detecting phishing attacks using natural language processing and machine learning. In 2018 ieee 12th international conference on semantic computing (icsc) (pp. 300-301). IEEE.

[3] Al-Yozbaky, R.S. and Alanezi, M., 2023, June. Detection and analyzing phishing emails using nlp techniques. In 2023 5th International Congress on Human-Computer Interaction, Optimization and Robotic Applications (HORA)(pp. 1-6). IEEE.

[4] Salloum, S., Gaber, T., Vadera, S. and Shaalan, K., 2022. A systematic literature review on phishing email detection sing natural language processing techniques. IEEE Access, 10, pp.65703-65727.

[5] Khare, A., Singh, S., Mishra, R., Prakash, S. and Dixit, P., 2022, March. E-Mail Assistant–Automation of E-Mail Handling and Management using Robotic Process Automation. In 2022 International Conference on Decision Aid Sciences and Applications (DASA) (pp. 511-516). IEEE.

[6] Chinnasamy, P., Krishnamoorthy, P., Alankruthi, K., Mohanraj, T., Kumar, B.S. and Chandran, L., 2024, March. AI Enhanced Phishing Detection System. In 2024 Third International Conference on Intelligent Techniques in Control, Optimization and Signal Processing (INCOS) (pp. 1-5). IEEE.

[7] Madhu Sudhan H V, 2019. Intelligent Email Extraction and Classification with NLP & Deep Learning. International Journal of Science and Research (IJSR), DOI: 10.21275/ART20196371.

[8] Anilkumar, C., Karrothu, A., Mouli, N.S. and Tej, C.B., 2023, January. Recognition and processing of phishing emails using NLP: A survey. In 2023 International Conference on Computer Communication and Informatics (ICCCI) (pp. 1-4). IEEE.

[9] Rabbi, M.F., Champa, A.I. and Zibran, M.F., 2023, May. Phishy? detecting phishing emails using ml and nlp. In 2023 IEEE/ACIS 21st International Conference on Software Engineering Research, Management and Applications (SERA) (pp. 77-83). IEEE.

[10] Egozi, G. and Verma, R., 2018, November. Phishing email detection using robust nlp techniques. In 2018 IEEE International Conference on Data Mining Workshops (ICDMW) (pp. 7-12). IEEE